PIC ネットワーク・ボード

取扱説明書

TR-67J60LAN/USB　V1.0

初版　　2009/09/21
合同会社 Trinity
http://www.itrinity.jp/

[TR-67J60LAN/USB]

目次

1. 製品の内容

本パッケージには、以下のものが含まれています。万一、不足などがございましたら、弊社宛にご連絡ください。

マイコンボード　TR-67J60LAN/USB　1
ジャンパー・ソケット：JP1 用【装着済み】　1

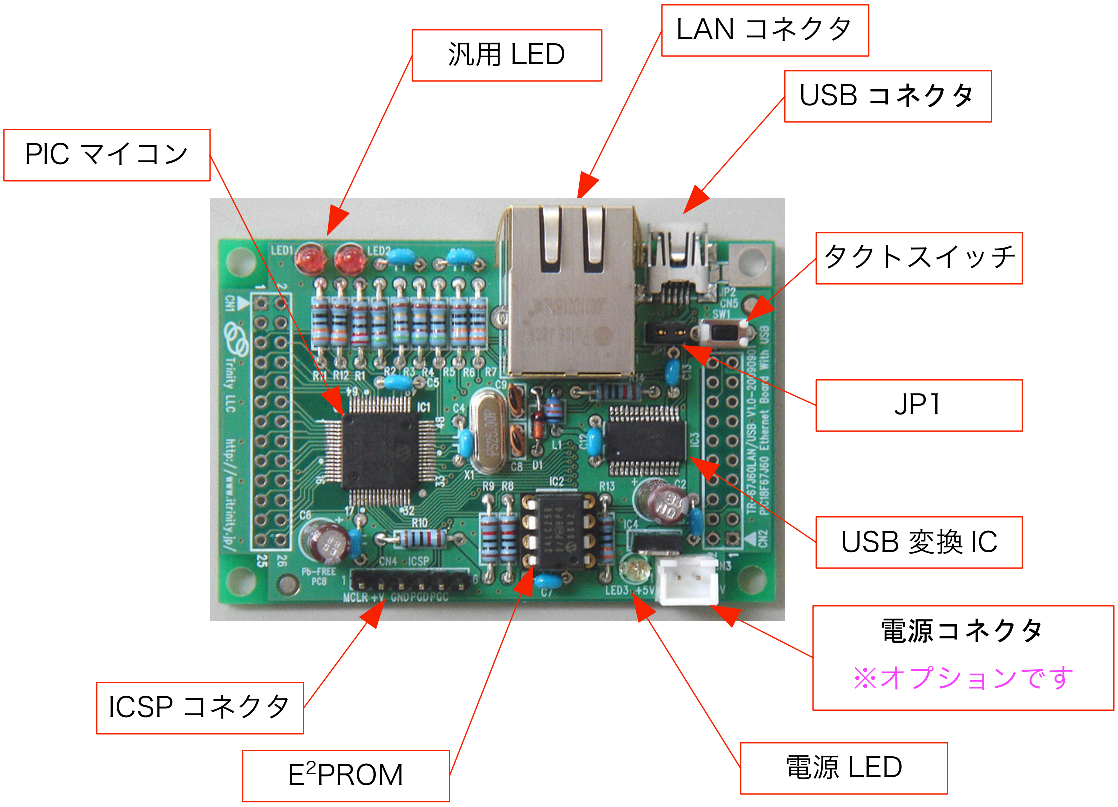
【連絡先】〒320-0844
栃木県宇都宮市菊水町 8-1　第 2 亀和田ビル 4 階　合同会社トリニティ
TEL　028-633-4343　FAX　028-633-5105

2. 仕様

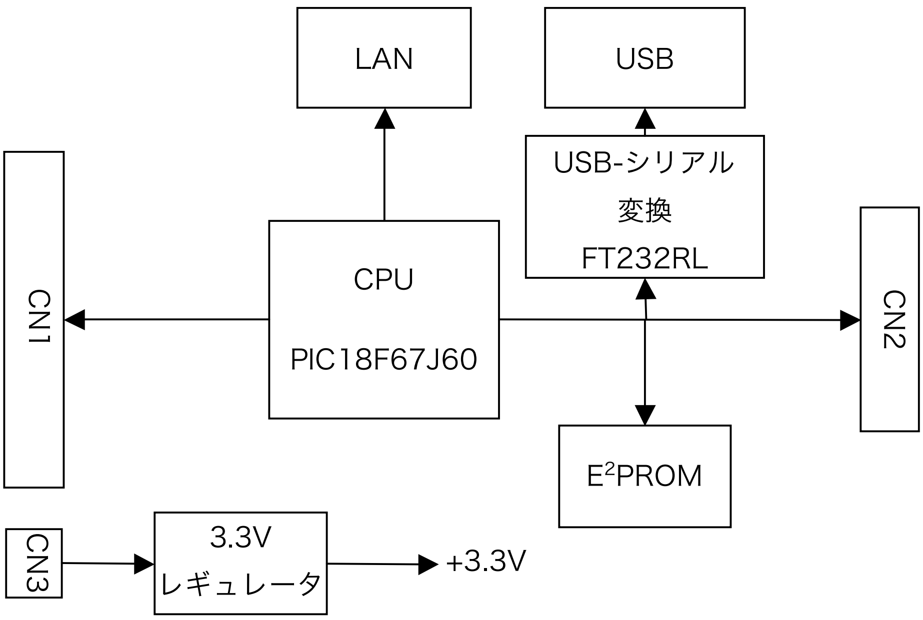
製品型番	TR-67J60LAN/USB
搭載 CPU	PIC18F67J60-I/PT
電源	DC5V
消費電流	約 170 mA
外形寸法	72mm×47mm
プリント基板	FR-4 ガラスエポキシ両面 t=1.6mm
ユーザーI/O	35（うち 2 本は USB 接続に使用済み）
クロック	25MHz 水晶発振子(動作クロックは 41.666MHz)
I/O コネクタ	26P×1　20P×1
LAN	10BASE-T　RJ-45 コネクタ
USB	USB1.1 ハイスピード　Mini-B コネクタ
ICSP コネクタ	6P ピンヘッダ
LED	電源表示×1　汎用×2
タクトスイッチ	1

3. 製品概要

3.1 各部の名称



3.2 ブロック図



3.3 電源入力

本ボードは、DC 5V単一電源または USB バスパワーで動作します。

外部から供給する 5V 電源は充分安定して、余裕のあるものをご用意ください。

・外部給電

電源は CN3 から供給してください。

ジャンパーJP1 は取り外してください。

または CN2 からも供給することができます。その際には GND はなるべく全ての PIN に接続してください。

・USB バスパワー給電(最大 400mA)

ジャンパーJP1 を取り付けてください。

USB ケーブルを接続し、パソコンから供給します。

パソコン本体の USB ポートに直接接続し、HUB を通しての接続はできるだけさけてください。

【注意】ジャンパーJP1 を取り付けた状態で外部電源と USB を両方接続しないでください。
内部回路が故障します。

3.4 動作電源

基板内部動作電圧は 3.3V です。

CPU の I/O 電源も 3.3V です。外部に回路を接続する場合には 3.3V 以上の電圧が加わらないようにご注意ください。

なお、CN1,CN2 には I/O 電源として 3.3V が出力されていますが、取り出せる電流は基板内部で消費される電流を含め約 300mA 程度です。

3.5 LAN コネクタ

10BASE-T 対応 Ethernet ポート・コネクタです。

ストレートケーブルで HUB と接続します。

パソコンと直接接続する場合にはクロスケーブルを使用してください。

3.6 USB コネクタ

USB1.1 対応 ミニ B タイプコネクタです。

パソコンからは仮想 COM ポートとして認識します。

3.7 LED

LED1	赤	汎用	RD1(59Pin)に接続
LED2	赤	汎用	RD2(58Pin)に接続
LED3	緑	電源表示	

3.8 タクトスイッチ

SW1　汎用スイッチ

RC5 に接続　10kΩでプルアップ

出荷時のプログラムでは IP アドレス変更用として機能します。

3.9 ICSP (In Circuit Serial Programming)コネクタ (CN4)

Microchip 社のデバッガ ICD2 や PICKit2 などを接続してプログラムの書き込み、デバッグを行います。

Pin	信号名	CPU ピン
1	MCLR	7
2	+3.3V	
3	GND	
4	PGD	37
5	PGC	42
6	NC	

4. ピン割り付け

4.1 CN1 ポート入出力(1)

Pin	信号名	CPU ピン	備考
1	GND		
2	GND		
3	+3.3V		
4	+3.3V		
5	RD2/CCP4/P3D	58	LED2
6	RD1/ECCP3/P3A	59	LED1
7	RD0/P1B	60	
8	RE5/P1C	61	
9	RE4/P3B	62	
10	RE3/P3C	63	
11	RE2/P2B	64	
12	RE1/P2C	1	
13	RE0/P2D	2	
14	RB0/INT0/FLT0	3	
15	RB1/INT1	4	
16	RB2/INT2	5	
17	RB3/INT3	6	
18	MCLR	7	
19	RG4/CCP5/P1D	8	
20	RF7/SS1	11	
21	RF6/AN11	12	
22	RF5/AN10/CVref	13	
23	RF4/AN9	14	
24	RF3/AN8	15	
25	RF2/AN7/C1OUT	16	
26	RF1/AN6/C2OUT	17	

4.2 CN2 ポート入出力(2)

Pin	信号名	CPU ピン	備考
1	GND		
2	GND		
3	+3.3V		
4	+3.3V		
5	RC3/SCK1/SCL1	34	I^2C SCL
6	RC4/SDI1/SDA1	35	I^2C SDA
7	RA3/AN3/Vref+	21	
8	RA2/AN2/Vref-	22	
9	RA5/AN4	27	
10	RA4/T0CKI	28	
11	RC1/T1OSI/ECCP2/P2A	29	
12	RC0/T1OSO/T13CKI	30	
13	RC6/TX1/CK1	31	FT232RL に接続済み
14	RC7/RX1/DT1	32	入出力としては使用不可
15	RC2/ECCP1/P1A	33	
16	RC5/SDO1	36	SW1 10kΩでプルアップ済
17	RB5/KBI1	43	
18	RB4/KBI0	44	
19	+5V		
20	+5V		

4.3 CN3 電源入力

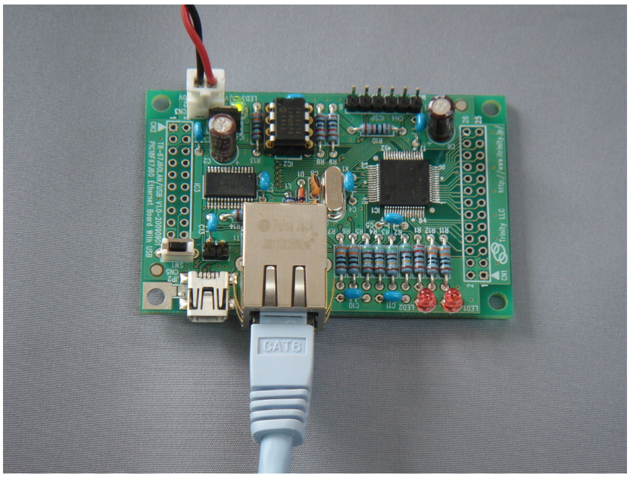
1	+5V
2	GND

5. 動作確認

5.1 接続

・LAN コネクタ：通常の LAN ケーブルを接続します。

接続方法 1　(USB を使用しない場合)

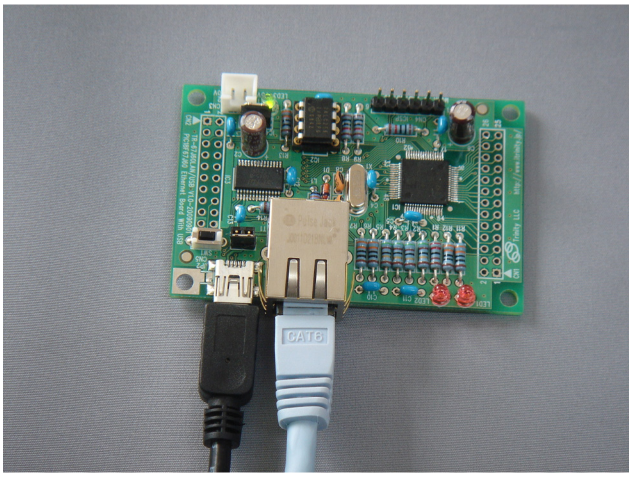


電源は 5V 電源を CN3 に接続してください。

(JP 1 は外してください)

【注意】ジャンパーJP1 を取り付けた状態で外部電源と USB を両方接続しないでください。
内部回路が故障します。

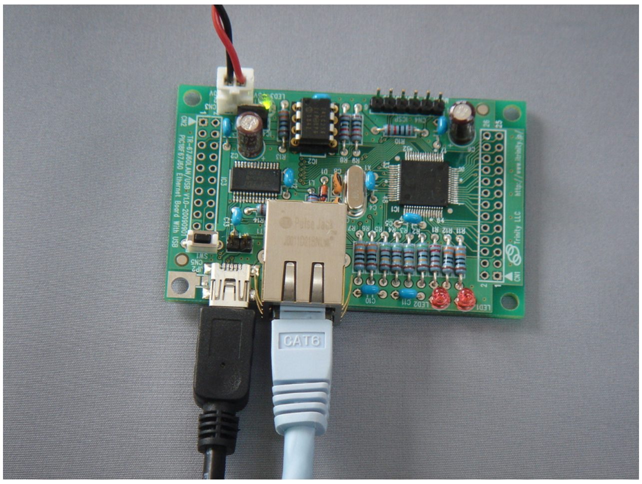
接続方法 2　(USB バスパワーを使用する場合)



電源は USB から供給されます。この場合、JP1 を取り付けてください。

USB は、パソコンから仮想 COM ポートとして認識しますので、UART にて通信可能となります。

接続方法 3　(USB バスパワーを使用しない場合)



JP1 を外し、5V 電源を CN3 に接続して、USB ケーブルを接続します。

USB は、パソコンから仮想 COM ポートとして認識しますので、UART にて通信可能となります。

【注意】ジャンパーJP1 を取り付けた状態で外部電源と USB を両方接続しないでください。
内部回路が故障します。

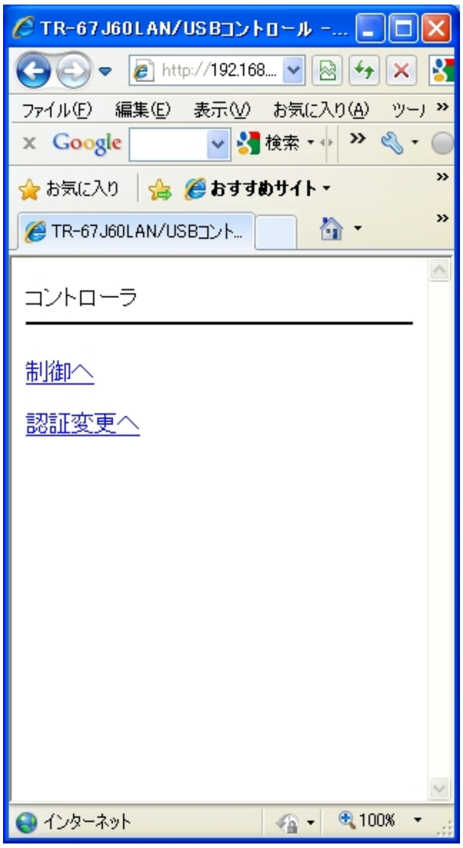
5.2 初期動作確認

出荷状態では弊社テストプログラムが書き込まれています。IP 取得に DHCP を参照しますので、LAN 内に DHCP サーバーが存在する環境で使用してください。

（ほとんどのブロードバンドルーターに DHCP サーバー機能があります）

DHCP サーバーがない場合や DHCP を固定する場合は、 6. IP アドレス変更 を参照してください。

LAN ケーブルを接続し、電源を供給してください。



ブラウザを起動します。

アドレス欄に「 http://trlanxxxx 」と入力します。

(LAN 内のみ) xxxx は MAC アドレス下位 2byte が入ります。

(このボードの LAN コネクタに貼ってある MAC アドレスの下位 2byte を入力してください)

すると左図の TOP ページが表示されます。

例　「00-10-47-FC-F9-22」と貼ってある場合は、「http://trlanf922」と入力します

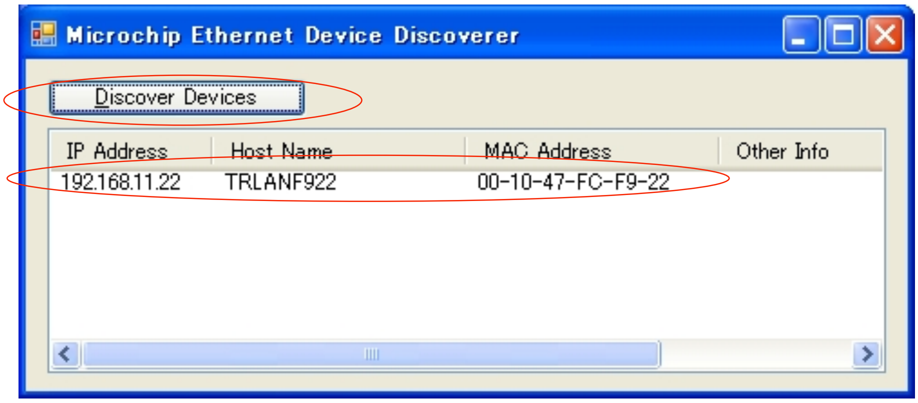
制御へ：制御画面へ移動します。(5.3 ブラウザの操作を参照)

認証変更へ：認証変更画面へ移動します。(5.3 ブラウザの操作を参照)

また、IP アドレスをブラウザのアドレス欄に入力して、上記の TOP ページを表示することも出来ます。IP アドレスを知る方法として Microchip 社の「Microchip Ethernet Discoverer」で確認できます。Microchip 社の TCP/IP Stack をインストールすると「C:\Microchip Solutions\Microchip\TCPIP Stack\Utilities」内にあります。

また、[ｽﾀｰﾄ]→[Microchip]→[TCPIP Stack v4.55]→[Ethernet Discoverer]からも起動できます。

TCP/IP Stack の入手とインストール方法については参考文献（「PIC で楽しむネットワーク接続機器の自作」　著者　後閑哲也氏 ）の 6-1(P102)を参照してください。

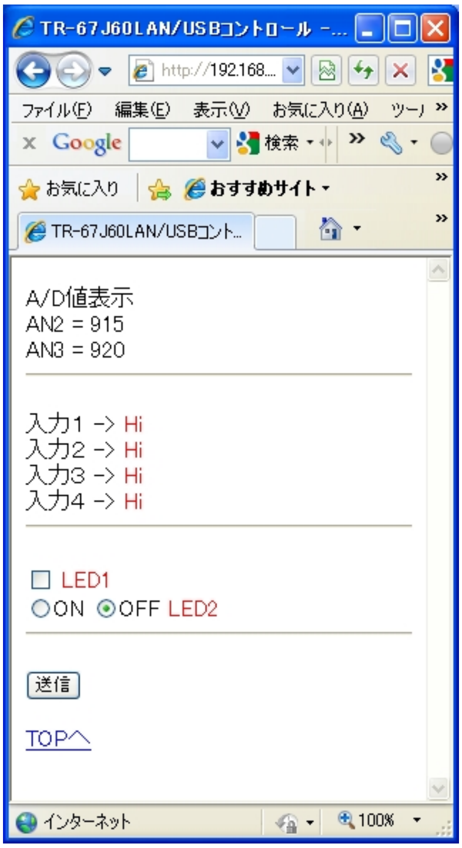


起動すると上図のように接続しているボード(TRLANxxxx)が表示され、MAC アドレスと割り当てられた IP アドレスが表示されます。xxxx は MAC アドレス下位 2byte が入ります。

何も表示されない場合は画面の上にある「Discover Devices」ボタンを押してください。ブラウザで表示したいボードをダブルクリックするか、ブラウザを起動してアドレス欄に、この場合は「 http://192.168.11.22 」と入力します。

5.3 ブラウザの操作

(制御画面)



AN2,3 には AN2 と AN3 の A/D 値を表示します。
(何も接続していない場合は不定な値を表示します)

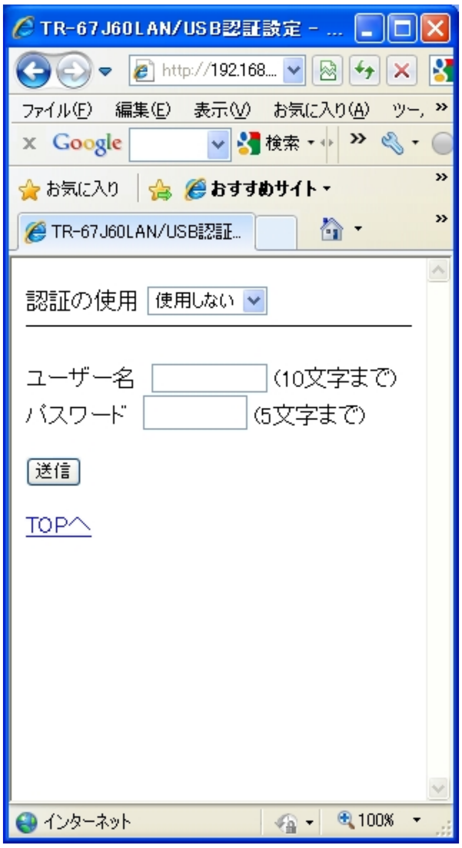
入力 1～4 は RB0～3 の電圧レベルを Hi/Lo で表示します。

LED1,2 はボード上の LED1,2 の ON/OFF をコントロールします。

LED1 はチェックボックスにチェックを付けると ON、チェックを外すと OFF になります。
LED2 はラジオボタンにて ON/OFF を選択します。

送信ボタンを押すことでこれらの情報を送信します。

(認証変更画面)



認証の使用は、初期状態では「使用しない」に設定してあります。認証の使用をする場合、「使用する」にして、ユーザー名、パスワードを設定してください。

ユーザー名は半角英数字 10 文字まで設定できます。

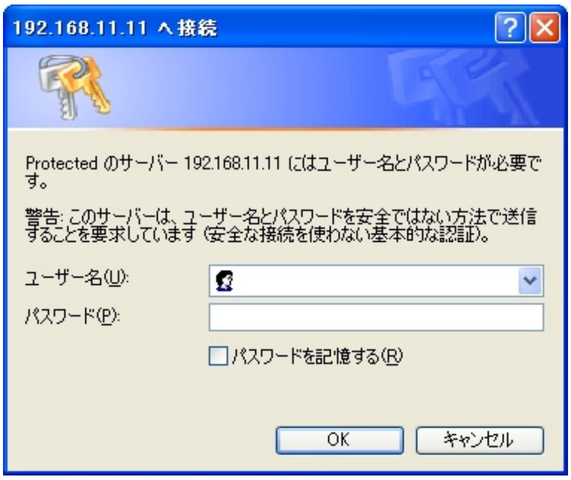
パスワードは半角英数字 5 文字まで設定できます。

送信は上記設定を有効にします

注：認証を使用する場合は、必ずユーザー名、パスワードを入力してください。空白の場合はエラーになります。

【お願い】
インターネットに接続する場合、セキュリティ保護の為できるだけ認証を使用するようにしてください。

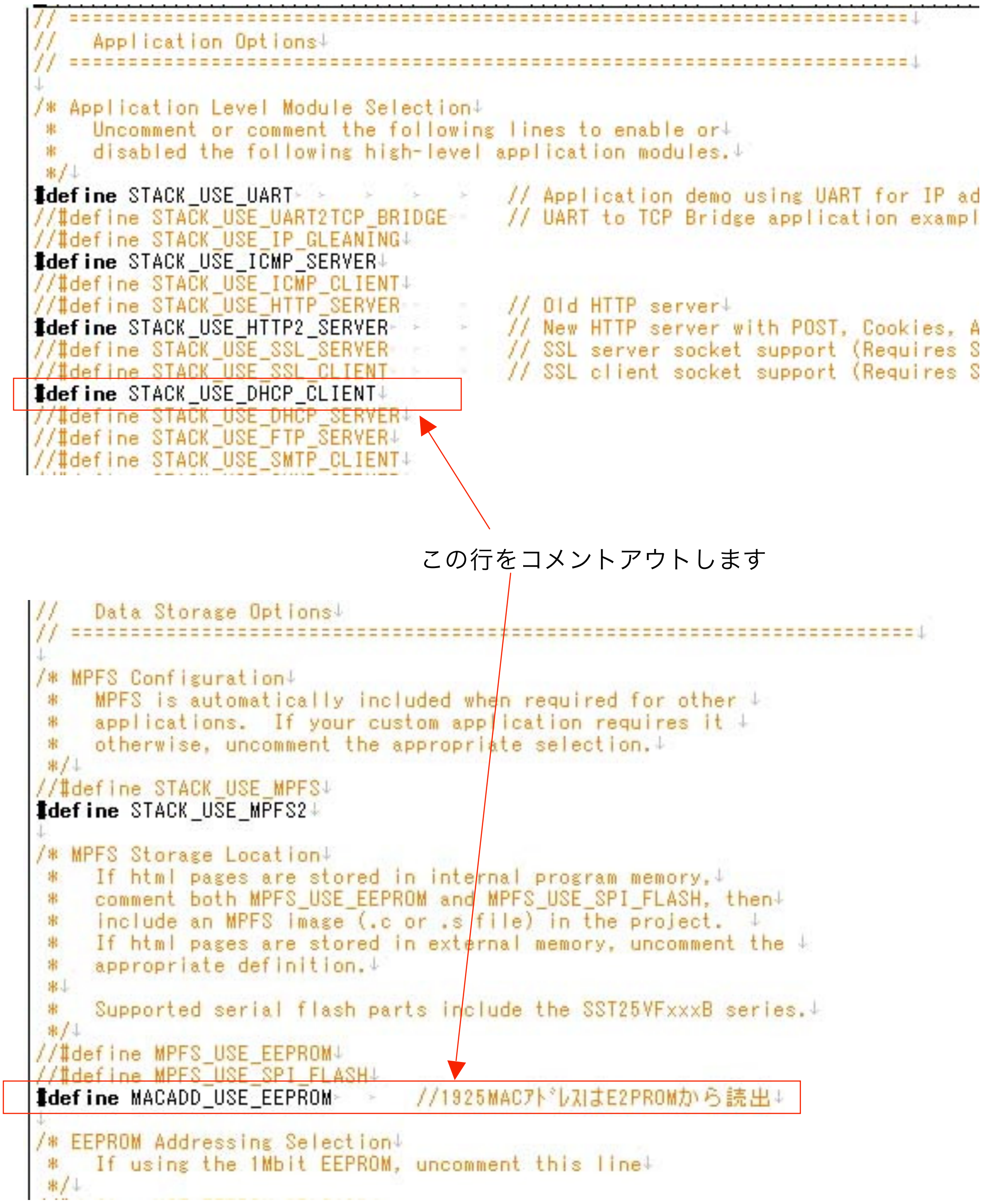
認証を使用する場合、TOP ページ以外の制御、認証変更ページには認証が必要となります。
下図のような画面が表示されます。(IE8 の場合)



6. IP アドレス変更

6.1 プログラムで変更（固定 IP アドレスにする）

TCPIPConfig.h 内 の #define STACK_USE_DHCP_CLIENT と #define MACADD_USE_EEPROM をコメントアウトします。MACADD_USE_EEPROM を定義することで、EEPROM(IC2)に MAC アドレス等の設定の書き込みと読出しを行っています。コメントアウトしないとプログラム上の設定が有効になりません。




```
// ========================================================================
//   Application Options
// ========================================================================

/* Application Level Module Selection
 *   Uncomment or comment the following lines to enable or
 *   disabled the following high-level application modules.
 */
#define STACK_USE_UART                  // Application demo using UART for IP ad
//#define STACK_USE_UART2TCP_BRIDGE     // UART to TCP Bridge application exampl
//#define STACK_USE_IP_GLEANING
#define STACK_USE_ICMP_SERVER
//#define STACK_USE_ICMP_CLIENT
//#define STACK_USE_HTTP_SERVER         // Old HTTP server
#define STACK_USE_HTTP2_SERVER          // New HTTP server with POST, Cookies, A
//#define STACK_USE_SSL_SERVER          // SSL server socket support (Requires S
//#define STACK_USE_SSL_CLIENT          // SSL client socket support (Requires S
#define STACK_USE_DHCP_CLIENT
//#define STACK_USE_DHCP_SERVER
//#define STACK_USE_FTP_SERVER
//#define STACK_USE_SMTP_CLIENT
```

```
//   Data Storage Options
// ========================================================================

/* MPFS Configuration
 *   MPFS is automatically included when required for other
 *   applications.  If your custom application requires it
 *   otherwise, uncomment the appropriate selection.
 */
//#define STACK_USE_MPFS
#define STACK_USE_MPFS2

/* MPFS Storage Location
 *   If html pages are stored in internal program memory,
 *   comment both MPFS_USE_EEPROM and MPFS_USE_SPI_FLASH, then
 *   include an MPFS image (.c or .s file) in the project.
 *   If html pages are stored in external memory, uncomment the
 *   appropriate definition.
 *
 *   Supported serial flash parts include the SST25VFxxxB series.
 */
//#define MPFS_USE_EEPROM
//#define MPFS_USE_SPI_FLASH
#define MACADD_USE_EEPROM       //1925MACｱﾄﾞﾚｽはE2PROMから読出

/* EEPROM Addressing Selection
 *   If using the 1Mbit EEPROM, uncomment this line
 */
```

下図は、TCPIPConfig.h 内の MAC アドレスや IP アドレスを宣言している部分です。
この変数がそのまま有効になります。

```
// =======================================================================
//   Network Addressing Options
// =======================================================================

/* Default Network Configuration
 *   These settings are only used if data is not found in EEPROM.
 *   To clear EEPROM, hold BUTTON0, reset the board, and continue
 *   holding until the LEDs flash.  Release, and reset again.
 */
#define MY_DEFAULT_HOST_NAME            "TR67J60LAN"

#define MY_DEFAULT_MAC_BYTE1            (0x00)
#define MY_DEFAULT_MAC_BYTE2            (0x10)
#define MY_DEFAULT_MAC_BYTE3            (0x47)
#define MY_DEFAULT_MAC_BYTE4            (0xFC)
#define MY_DEFAULT_MAC_BYTE5            (0xF9)
#define MY_DEFAULT_MAC_BYTE6            (0x22)

#define MY_DEFAULT_IP_ADDR_BYTE1        (192ul)
#define MY_DEFAULT_IP_ADDR_BYTE2        (168ul)
#define MY_DEFAULT_IP_ADDR_BYTE3        (1ul)
#define MY_DEFAULT_IP_ADDR_BYTE4        (105ul)

#define MY_DEFAULT_MASK_BYTE1           (255ul)
#define MY_DEFAULT_MASK_BYTE2           (255ul)
#define MY_DEFAULT_MASK_BYTE3           (255ul)
#define MY_DEFAULT_MASK_BYTE4           (0ul)

#define MY_DEFAULT_GATE_BYTE1           (192ul)
#define MY_DEFAULT_GATE_BYTE2           (168ul)
#define MY_DEFAULT_GATE_BYTE3           (1ul)
#define MY_DEFAULT_GATE_BYTE4           (1ul)

#define MY_DEFAULT_PRIMARY_DNS_BYTE1    (192ul)
#define MY_DEFAULT_PRIMARY_DNS_BYTE2    (168ul)
#define MY_DEFAULT_PRIMARY_DNS_BYTE3    (1ul)
#define MY_DEFAULT_PRIMARY_DNS_BYTE4    (1ul)

#define MY_DEFAULT_SECONDARY_DNS_BYTE1  (192ul)
#define MY_DEFAULT_SECONDARY_DNS_BYTE2  (168ul)
#define MY_DEFAULT_SECONDARY_DNS_BYTE3  (1ul)
#define MY_DEFAULT_SECONDARY_DNS_BYTE4  (1ul)
```

MAC アドレス宣言部分

IP アドレス宣言部分

プログラムを変更した場合、コンパイルし、書き込みをしてください。
(CN4 が ICSP コネクタになっています)

※プログラム開発に必要なもの

C コンパイラと統合環境

無償で Microchip 社の web ページからダウンロードできます。

書き込み器(ICSP 対応)

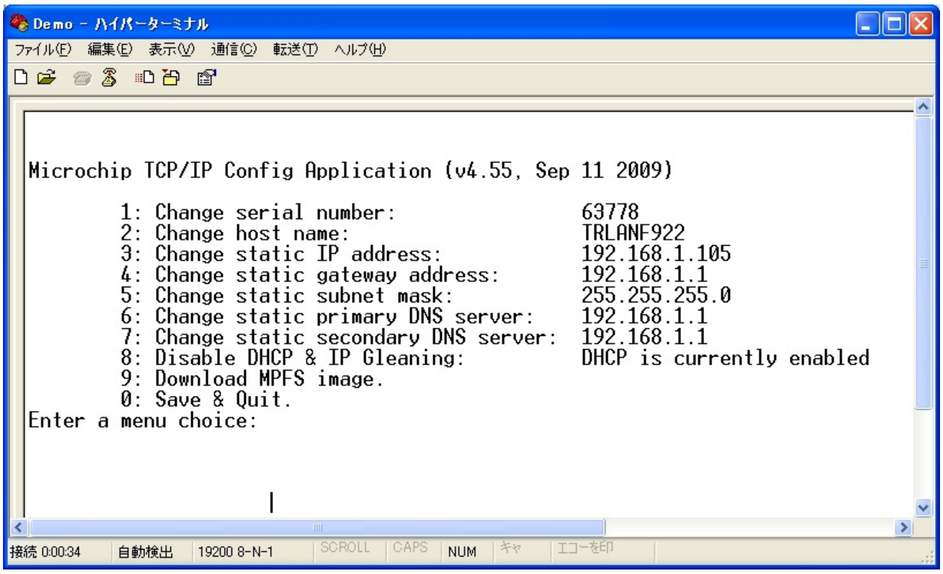
ICD2,ICD3 と PICkit2 があります。

(ボード上の ICSP コネクタと書き込み器を接続するケーブルは別途ご用意ください。)

6.2 シリアル通信で変更

通信条件（19200bps,8bit,パリティなし,ストップビット 1,フロー制御なし）

USB ケーブルにて PC と接続し、ターミナルソフトを起動します。SW1 を押しながら電源を入れます。(USB ケーブルより電源を供給している場合は JP1 を一度はずし、SW1 を押しながら JP1 を取り付けてください) SW1 を離すと下図の設定画面が表示されます。



設定方法

1～9 を選択し入力します。次に変更後の数値を入力し、「Enter」を押します。

最後に「0」を押すと EEPROM に書き込み、「Settings saved.」と表示され通常の動作に戻ります。

MAC アドレスの変更の方法

MAC アドレスは、「1」の Change serial number なので、「1」を入力します。

次に MAC アドレスを入力しますが、変更できるのは MAC アドレスの下位 2byte のみです。下位 2byte の 16 進数を 10 進数(0～65535 の範囲)にして入力し、「Enter」を押します。

例　MAC アドレスが 00-10-47-FC-F9-22 と変更する場合

下位 2byte の F9-22 に変更します。16 進数の F922 を 10 進数に変換すると 63778 になります。「1」を入力後、「63778」と入力します。「Enter」を入力すると変更後の値が表示されます。

【注意】

MAC アドレスの上位 4byte はプログラム内の値を使用します。弊社出荷時のプログラム以外のプログラムを弊社のボードで動作させる場合、MAC アドレスはボードに記載してある値をプログラム内に記述してください。

任意の IP アドレスにする場合

「3」を入力します。この後任意の IP アドレスを入力します。

例　192.168.11.50 に変更する場合

「3」を入力します。次に「192.168.11.50」を入力し、「Enter」を入力すると変更後の値が表示されます。

固定 IP アドレスにする場合

「8」を入力します。（出荷状態では DHCP サーバー参照となっています。「8」を入力すると固定IPアドレスになります。またこの状態からもう一度「8」を入力するとDHCPサーバー参照となります。）

全ての変更終了後に「0」を入力し、EEPROM に今回設定した値を書き込み通常の動作に戻ります。

7. 参考プログラム

7.1 追加回路

参考文献（「PIC で楽しむネットワーク接続機器の自作」 著者 後閑哲也氏）の第 6 章（6-1～6-7)、11 章(11-3)のプログラムを動作させせることができます。TR67J60LAN ボード上には LED が 2 個なので LED を 4 個、スイッチを 3 個、LCD を 1 個、アナログ入力 2 点を追加すると動作確認が簡単に出来ます。

(製作には第 5 章の P91 と弊社ボードの回路図を参考にしてください。)

7.2 変更箇所(6-1～6-5,11-3)

追加の回路を製作した場合、プログラムはそのまま動作します。TR67J60LAN ボードのみで動作確認する場合は、プログラムを変更する必要があります。

"HardwareProfile.h"のピン設定の変更

```
#elif defined(YOUR_BOARD)
// Define your own board hardware profile here
	// I/O pins
	#define LED0_TRIS		(TRISDbits.TRISD1)	//RE2 -> RD1
	#define LED0_IO			(PORTDbits.RD1)
	#define LED1_TRIS		(TRISDbits.TRISD2)	//RE3 -> RD2
	#define LED1_IO			(PORTDbits.RD2)
	#define LED2_TRIS		(TRISEbits.TRISE4)
	#define LED2_IO			(PORTEbits.RE4)
	#define LED3_TRIS		(TRISEbits.TRISE5)
	#define LED3_IO			(PORTEbits.RE5)
	#define LED4_TRIS		(TRISDbits.TRISD0)
	#define LED4_IO			(PORTDbits.RD0)
	#define LED5_TRIS		(TRISEbits.TRISE2)	//RD1 -> RE2
	#define LED5_IO			(PORTEbits.RE2)
	#define LED6_TRIS		(TRISEbits.TRISE3)	//RD2 -> RE3
	#define LED6_IO			(PORTEbits.RE3)
	#define LED7_TRIS		(TRISEbits.TRISE0)
	#define LED7_IO			(PORTEbits.RE0)
	#define LED_IO			(*((volatile unsigned char*)(&PORTE)))

	#define BUTTON0_TRIS		(TRISBbits.TRISB3)
	#define BUTTON0_IO		(PORTBbits.RB3)
	#define BUTTON1_TRIS		(TRISBbits.TRISB2)
	#define BUTTON1_IO		(PORTBbits.RB2)
	#define BUTTON2_TRIS		(TRISBbits.TRISB1)
	#define BUTTON2_IO		(PORTBbits.RB1)
	#define BUTTON3_TRIS		(TRISBbits.TRISB0)
	#define BUTTON3_IO		(PORTBbits.RB0)
	#define BUTTON_SET_TRIS	(TRISBbits.TRISB4)
	#define BUTTON_SET_IO		(PORTBbits.RB4)

	// LCD I/O pins
	#define LCD_DATA_TRIS		(TRISF)
	#define LCD_DATA_IO		(LATF)
	#define LCD_RD_WR_TRIS	(TRISFbits.TRISF1)
	#define LCD_RD_WR_IO		(LATFbits.LATF1)
	#define LCD_RS_TRIS		(TRISFbits.TRISF2)
	#define LCD_RS_IO		(LATFbits.LATF2)
	#define LCD_E_TRIS		(TRISFbits.TRISF3)
	#define LCD_E_IO		(LATFbits.LATF3)
```

LED0_IO を LED1 に、LED1_IO を LED2 に割り当てますので、上図のように変更します。

TR67J60LAN ボードの LED1 が 0.5 秒間隔で点滅動作をします。

7.3 変更箇所(6-6,6-7)

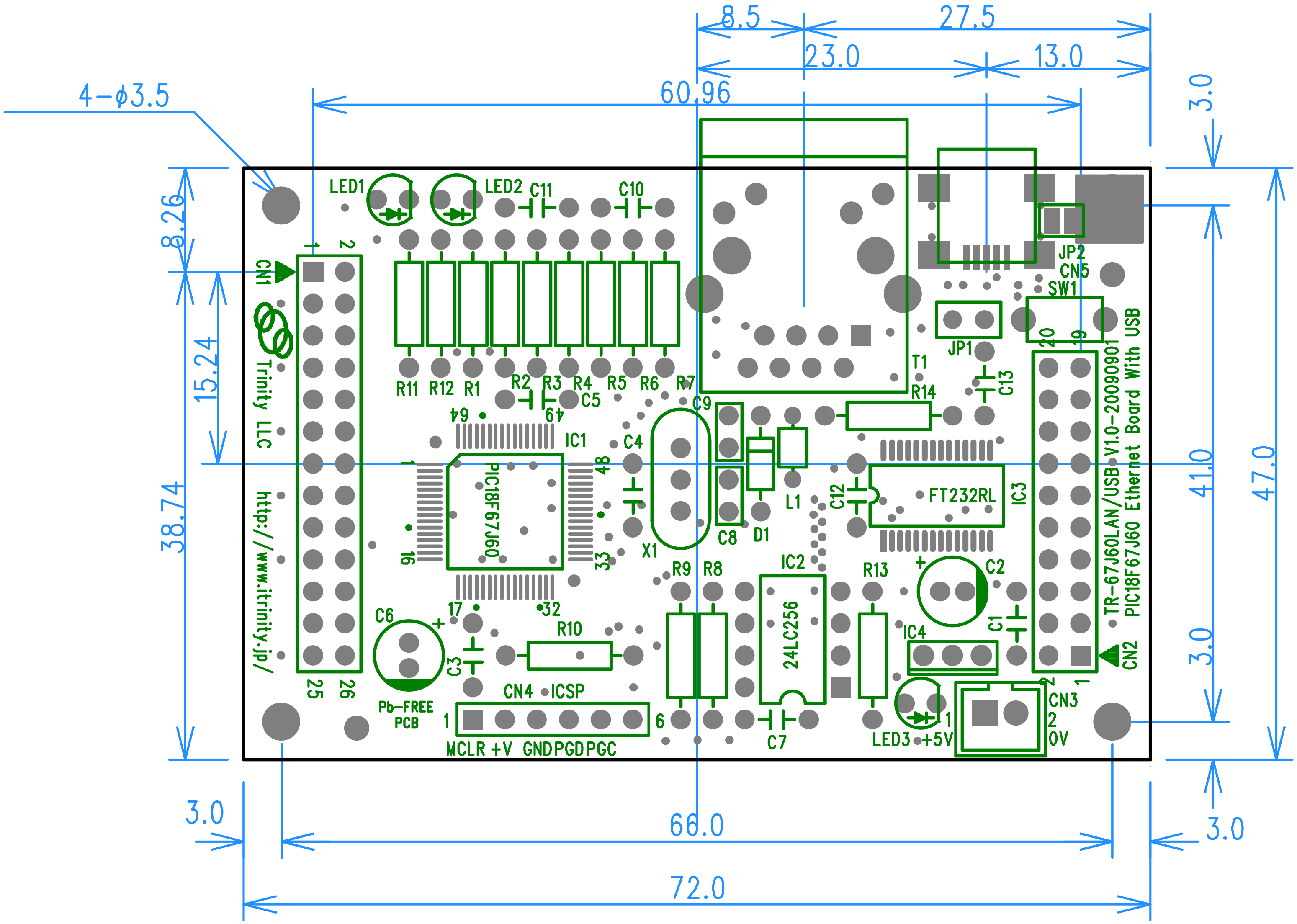
追加の回路を製作した場合、プログラムはそのまま動作します。TR67J60LAN ボードのみで動作確認する場合は、"HardwareProfile.h"のピン設定を変更します。

```
#elif defined(YOUR_BOARD)
// Define your own board hardware profile here
    // I/O pins
    #define     LED0_TRIS       (TRISEbits.TRISE2)
    #define     LED0_IO         (PORTEbits.RE2)
    #define     LED1_TRIS       (TRISEbits.TRISE3)
    #define     LED1_IO         (PORTEbits.RE3)
    #define     LED2_TRIS       (TRISDbits.TRISD1)      //RE4 -> RD1
    #define     LED2_IO         (PORTDbits.RD1)
    #define     LED3_TRIS       (TRISDbits.TRISD2)      //RE5 -> RD2
    #define     LED3_IO         (PORTDbits.RD2)
    #define     LED4_TRIS       (TRISDbits.TRISD0)
    #define     LED4_IO         (PORTDbits.RD0)
    #define     LED5_TRIS       (TRISEbits.TRISE4)      //RD1 -> RE4
    #define     LED5_IO         (PORTEbits.RE4)
    #define     LED6_TRIS       (TRISEbits.TRISE5)      //RD2 -> RE5
    #define     LED6_IO         (PORTEbits.RE5)
    #define     LED7_TRIS       (TRISEbits.TRISE0)
    #define     LED7_IO         (PORTEbits.RE0)
    #define     LED_IO          (*((volatile unsigned char*)(&PORTE)))

    #define     BUTTON0_TRIS        (TRISBbits.TRISB3)
    #define BUTTON0_IO      (PORTBbits.RB3)
    #define     BUTTON1_TRIS        (TRISBbits.TRISB2)
    #define BUTTON1_IO      (PORTBbits.RB2)
    #define     BUTTON2_TRIS        (TRISBbits.TRISB1)
    #define BUTTON2_IO      (PORTBbits.RB1)
    #define     BUTTON3_TRIS        (TRISBbits.TRISB0)
    #define BUTTON3_IO      (PORTBbits.RB0)
    #define     BUTTON_SET_TRIS (TRISBbits.TRISB4)
    #define BUTTON_SET_IO       (PORTBbits.RB4)

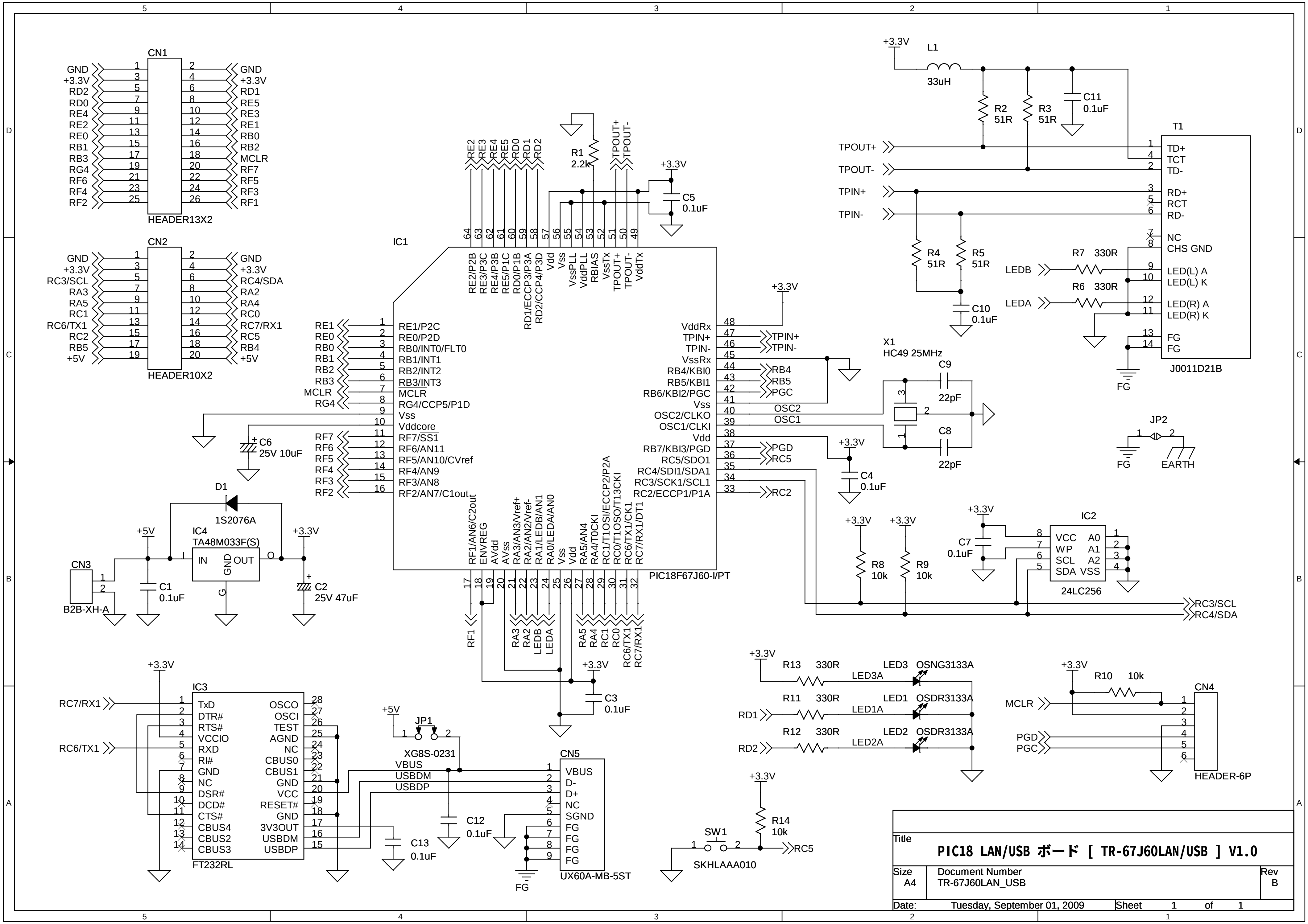
    // LCD I/O pins
    #define     LCD_DATA_TRIS       (TRISF)
    #define     LCD_DATA_IO     (LATF)
    #define     LCD_RD_WR_TRIS  (TRISFbits.TRISF1)
    #define     LCD_RD_WR_IO        (LATFbits.LATF1)
    #define     LCD_RS_TRIS     (TRISFbits.TRISF2)
    #define     LCD_RS_IO       (LATFbits.LATF2)
    #define     LCD_E_TRIS      (TRISFbits.TRISF3)
    #define     LCD_E_IO            (LATFbits.LATF3)
```

6-6、6-7 のプログラムでは LED2,3 を PC より制御しますので、そこに TR67J60LAN ボードの LED1,LED2 を割り当てています。よって、点滅動作をする LED はなくなります。



PIC18F67J60 LAN/USB Board V1.0 2009/09/01

部品面 部品面シルク



Title	**PIC18 LAN/USB ボード [TR-67J60LAN/USB] V1.0**	
Size A4	Document Number TR-67J60LAN_USB	Rev B
Date: Tuesday, September 01, 2009	Sheet 1 of 1	